Basant Khalil

614-717-8842 | bak94@cornell.edu | linkedin | github

Education

CORNELL UNIVERSITY: *Summa Cum Laude* Ithaca, NY
B.S. Computer Science & Electrical and Computer Engineering, GPA: 4.041 *Aug. 2020 – May 2023*
IEEE-Eta Kappa Nu (IEEE-HKN), the International Honor Society of Electrical and Computer Engineering Cornell Chapter.

DUBLIN JEROME HIGH SCHOOL Columbus, OH
High School Honor Diploma; Valedictorian; GPA: 5.0 *Aug. 2018 – May 2020*

Experience

Full Stack Engineering Intern May 2023 – Present
Hewlett Packard Enterprise *Durham, NC*

Cornell Undergraduate Research Assistant, Capra Research Group May 2023 – Present
Cornell University, Advisor: Adrian Sampson *Remote*

- Worked on Calyx, a compiler infrastructure for languages targeting hardware accelerators which automatically optimizes and transformes programs into synthesizable hardware designs.
- Utilized Calyx's control language to simplify the encoding of high-level semantics.

Analysis of Algorithms (CS 4820) Teaching Assistant Jan 2023 – Present
Discrete Structures (CS 2800) Teaching Assistant Aug 2022 – Dec 2022
Circuits for Electrical and Computer Engineers (ECE 2100) HKN Tutor Aug 2022 – Dec 2022
Cornell University *Ithaca, NY*

- Led discussions and provided individualized support to students both inside and outside office hours.
- Graded assignments and exams and provided feedback to students.

Calculus for Engineers (Math 1910) Course Assistant Aug 2021 – Dec 2021
Physics I: Mechanics & Heat (Phys 1112) ELI Peer Tutor Aug 2021 – Dec 2021
Linear Algebra for Engineers (Math 2940) ELI Peer Tutor Aug 2021 – Dec 2021
Cornell University *Ithaca, NY*

- Worked with small groups of students to supplement classroom instruction.
- Created and reviewed practice problems and test materials.
- Worked with the section TA to facilitate group work on workshop problems for 6+ recitation sections.

Software Engineer Intern Mar 2021 – June 2021
JP Morgan Chase *Remote*

- Navigated JP Morgan Chase's frameworks to efficiently develop & analyze data sets
- Leveraged technical skills to help implement systems that enabled accurate data visualization
- Identified & resolved bugs using multiple programming languages, including Java

Data Intern June 2021 – Aug 2021
AXA Equitable *Remote*

- Developed advanced statistical models and algorithms for forecasting & predicting customer behaviors
- Utilized large databases and professional statistical techniques to collect, organize and interpret data from customers

Technical Skills

Languages: Java, Python, C/C++, SQL (Postgres), JavaScript, HTML/CSS, Matlab, Verilog, VHDL
Developer Tools: Git, VS Code, Visual Studio, Atom, IntelliJ, Eclipse
Libraries: pandas, NumPy, Matplotlib, PyTorch

Awards

- Cornell University: Dean's List (all semesters)
- Cornell University: Summa Cum Laude
- IEEE-Eta Kappa Nu, the International Honor society of Electrical and Computer Engineering Cornell Chapter
- Tau Beta Pi, the National Engineering Honor Society Cornell Chapter, member
- Richard A. Tapia Award for significant leadership in CS: Cornell University
- President's Education Award for Educational Excellence
- Dublin Jerome High School: Summa Cum Laude; Honor/Merit Roll; Science & Technology Seal; Seal of Biliteracy
- AP Scholar With Distinction

Projects

Class Roster Database | *Ocaml, OOP* Apr 2022 – May 2022

- Parsed data from Cornell Class roster.
- Outputted a roadmap to fulfill one or more majors inputted by the user along with minors supporting 10+ majors.
- Provided relevant information regarding a certain class or more inputted by the user, such as the time it occurs, professor's names, the mode of delivery, etc. from the database incorporating 100+ classes.
- Allowed the user to calculate his GPA inputting his classes along with the corresponding final grades and explore a variety of classes occurring at some time frame for 100+ classes.

Blocking Cache | *Verilog, Python, C/C++* Oct 2021 – Nov 2021

- Implemented an FSM cache microarchitecture with a hit latency of 4 cycles, handling latency insensitive interfaces, init transaction, and eviction responses.
- Used an incremental design approach to implement the init transaction, read hit path, write hit path, refill path, evict path, and banking support.
- Developed a two-way set-associative, write-back, write-allocate cache with a capacity of 256 bytes and a cache line size of 16 bytes, using a least-recently-used (LRU) replacement policy to choose between both ways during eviction.
- Designed an init transaction to write data into the cache without a refill from main memory, simplifying testing and improving cache model functionality.

Single-Core and Multi-Core Systems | *Verilog, Python, C/C++* Nov 2021 – Dec 2021

- Designed and implemented a cache network and a memory network, each consisting of request/response network pairs, for interconnecting processors, caches, and main memory.
- Developed a multi-core data cache consisting of the cache network, 4 cache banks, and the memory network.
- Successfully composed four routers to create the complete ring network and put together the complete multi-core system, including Cache Network, Memory Network, and Multi-Core Data Cache.
- Implemented a new ring network with 4 input and output stream interfaces to interconnect the processors to the data caches and main memory interfaces, using NetRouterRouteUnit.v and NetRouterSwitchUnit.v to create a network router with three normal queues.

Pipelined Processor | *Verilog, Python, C/C++* Sep 2021 – Oct 2021

- Designed and implemented two five-stage pipelined microarchitectures for the TinyRV2 instruction set architecture using stalling and bypassing techniques to avoid hazards.
- Utilized various design principles, including modularity, hierarchy, and encapsulation, and design patterns such as message interfaces, control/datapath split, and pipelined control.
- Implemented a wide range of instructions, including CSR, Reg-Reg, Reg-Imm, Memory, Jump, and Branch instructions, using Verilog HDL for the RISC-V instruction set architecture (ISA).
- Leveraged the latency-insensitive stream interfaces implemented using the val/rdy microprotocol for processor interface and integrated them into the processor design, leading to an overall improvement in processor performance.

Iterative Integer Multiplier | *Verilog, Python, C/C++* Aug 2021 – Sep 2021

- Designed and implemented two implementations of an iterative integer multiplier: a fixed-latency implementation and a variable-latency implementation, using Verilog hardware description language.
- Demonstrated understanding of abstraction levels including functional- and register-transfer-level modeling, design principles including modularity, hierarchy, and encapsulation, and design patterns including streaming interfaces, control/datapath split, and FSM control.
- Demonstrated proficiency in modular design principles, including the control/datapath split design pattern, by decomposing the baseline design into two separate modules: the datapath and the control unit.

Maze Mapping Robot | *C/C++* Sep 2021 – Dec 2021

- Design/implement an algorithm to enable the robot to navigate a maze to find paths, walls, and some treasures, consisting of a dim near infrared LED blinking at approximately 25 kHz .
- Build the physical system to enable it to communicate information to a base station graphically displaying the progress.
- Work on designing the circuitry to show on the monitor where the treasures were found as well as manage the LEDs on the robot to turn on/off when desired with desired colors.

Relevant Courses

Analysis of Algorithms, Operating Systems, Embedded Systems, Computer Architecture, Intelligent Physical Systems, Functional Programming, Robotics, Database Systems, Computer Networks, Computer Vision